ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.03 No.03 (2025)

COMPARATIVE STUDY OF REQUIREMENTS PRIORITIZATION TECHNIQUES

***Hamera bibi*[1] *, Aneeqa Rauf*[2] *, Khalid Hamid*[3]**

Hamera bibi
Department of Software Engineering,
SuperiorUniversity, Lahore,
Punjab, Pakistan;
Corresponding Author *Email: hamerabibi6@gmai.com*

Aneeqa Rauf
Department of Software Engineering,
SuperiorUniversity, Lahore,
Punjab, Pakistan;
Corresponding Author *Email: raufaneeqa920@gmail.com*

Khalid Hamid
Department of Computer Science and Information Technology,
SuperiorUniversity, Lahore,
Punjab, Pakistan;
Corresponding Author *Email: khalid6140@gmail.com*

Abstract:

In software engineering, requirements prioritization is a critical stage that ensuresthe timely distribution of highvalue structures while working with restricted resources. In this study, 19 popular requirements prioritization policies are associated, and their efficiency is studied from a range of approaches, counting scalability, accuracy, shareholder contribution, and related flexibility. The goal is to define which technique or techniques are best suitable for a particular project established on dynamics like team size, complexity of requirements, time, cost, and risk.

This comparative study's main goal is to evaluate both modern techniques relating Artificial Intelligence, Fuzzy Logic, and Huge Language Models (LLMs) and more conservative methods such as Analytic Hierarchy Process (AHP), MoSCoW, Planning Game, and Value-Oriented Prioritization (VOP). The rewards of each method are examined concluded experiential case studies, which authenticate that LLM-based tools distribute automation and speed in Agile atmospheres, while AHP is dependable and dependable precisely.

The procedure involved collecting and observing available research from 2015 to 2025 with an importance on relative presentation, shareholder satisfaction, and choice accuracy. Surveys, models, and relative testing were inspected to control the prizes and drawbacks of each process. The results show that hybrid mockups, such as restraint solvers, fuzzy AHP, and AI-assisted methods, perform better than outdated ones in relations of flexibility and usability, particularly in vibrant or important tasks.

The revision proposes a context-aware method for operation, utilizing fusion/AI mockups where robotics and scalability are critical, MoSCoW for Agile groups, and AHP for systematic stability. In software development and condition manufacturing measures, this research supports in constructing well-read results.

Keywords:

Prioritization – Techniques – Scalability – Stakeholders – Automation **Introduction**:

Toward assurance that the maximum significant and crucial structures are providing within reserved resources and growth series, well-organized supplies arranging is critical in modern software engineering. Prioritization supports teams in systematically measuring and selecting requirements established on their rank, feasibility, commercial worth, and arrangement with planned objects in bright of the rising difficulty of software systems and the unstable stresses of shareholders. Nevertheless, the manufacturing continues to face a problematic: however, there are an amount of diverse prioritizing plans, there isn't any planned information on which

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR

method is best for a given condition. Although they deliver well-established outlines for prioritization, outdated methods such as the Analytic Hierarchy Process (AHP), MoSCoW, Planning Game, and Value-Oriented Prioritization (VOP) frequently fail when scalability, adaptability, or automation are needed[1] [2]

These methods may not be flexible sufficient for projects with continually unstable requirements or various shareholder input because they are classically physical and trust on expert judgment. Concurrently, new approaches that make use of AI, fuzzy logic, and large language models (LLMs) have exposed potential in civilizing prioritization measures through data-driven insights, dynamic learning, and automation[3] . These modern approaches current both new possibilities and problems in terms of interpretability, difficulty, and integration.[4] Software sides still have worry selecting ordering plans that fit their project needs, team dynamics, and practical limitations, even though there are many different methods accessible. Decisions about priorities are still normally made casually or without taking long-term effects into account, which consequences in less than ideal feature delivery and lower shareholder fulfillment. A combined, relative model that supports in understanding the trade-offs between conservative and AI-driven methods is missing. Additionally, the proportional research that is currently accessible in educational and industrial literature has not reserved up with the quick development of intellectual tools and hybrid models [1]. Teams normally deficiency the training essential to appropriately use the latest tools or are unaware of them.

Without attractive into version real-world application restrictions like shifting shareholder priorities, time-sensitive decision-making, team size, economic constraints, or the accessibility of associate tools, the popular of earlier studies either associate a small number of approaches or concentrate on a single method. [5] Additionally, there hasn't been much research done on including modern methods like hybrid methods and AI-powered tools into organized assessment frameworks. For practitioners and scholarly researchers who must understand not only how each technique functions but also when and where it is most effective, this gap breeds doubt [6][8]. It also prevents companies from using evidence-based prioritization techniques to optimize their software growth cycles.

By execution a thorough contrast of 19 requirement prioritization approaches, ranging from outmoded to AI-based and hybrid models, this study strive for to address the aforementioned issues. The three main objectives of the study are to: (1) determine which techniques are most useful in several project contexts, such as Agile versus traditional environments; (2) assess the ease of implementation of each technique based on team size, decision-making structure, available tools, and shareholder contribution; and (3) propose suitable techniques for realworld situations using a framework that is both practical and flexible. In addition to outlining the welfares and disadvantages of each method, this multi-layered analysis highlights the background fit necessary for effective implementation.

Two research questions are set forward to direct this study: which requirement prioritization approaches offer the best trade-off between precision, scalability, and usability for modern software development projects? And how do hybrid and AI-enhanced prioritization models stack up against conservative techniques in terms of flexibility and implementation productivity?

The study collected and analyzed 19 peer-reviewed studies and academic articles printed between 2015 and 2025 in order to achieve these aims. Through systematic searches of openaccess sources like arXiv, a, ResearchGate, and ScienceDirect, techniques were found. Scalability, accuracy, shareholder involvement, tool support, learning curve, cost, and

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR

automation capabilities were among the important evaluation criteria that were used to assess each approach.
Strengths, drawbacks, and optimal usage scenarios were highlighted using relative matrices and decision-making models. Additionally, thematic analysis and qualitative synthesis were conducted to find recurring themes, new trends, and related proposals among various research aids. [1] [2].
By submission a joined, current framework that connects conservative requirement prioritization models with new intelligent methods, this study advances the field of software engineering. It gives development teams, project managers, and business specialists practical visions to choose appropriate approaches based on practical requirements. Additionally, it highlights the importance of flexible and scalable prioritization in dynamic settings, providing references that facilitate both academic research and real-world application.
This study proposes to improve the arrangement between prioritization theory and its implementation by undertaking these fundamental problems, opening the door for software development methods that are more effective, value-driven, and shareholder-centered.

1.1. INTRODUCTION DIAGRAM

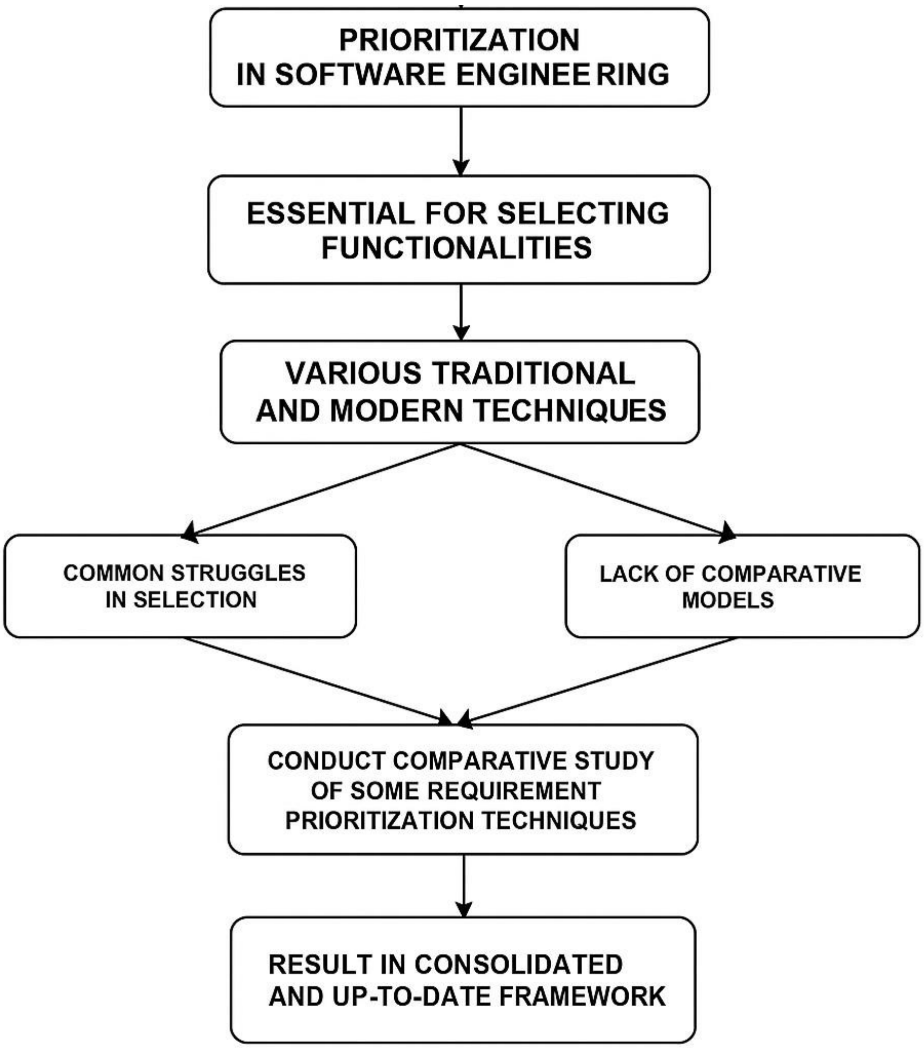

Figure 1: Introduction Diagram of Prioritization Technique Framework

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR

The consequence of setting significances in software engineering to select essential structures is demonstrated by this flowchart. It draws consideration to the application of both conservative and modern methods as well as the shared difficulty in selecting the best one. A combined and updated outline is produced as a result of the study's comparative analysis. **Literature Review**

For several years, software engineering has absorbed deeply on requirements prioritization, and many approaches have been developed to help rank software requirements according to criteria like value, risk, cost, and shareholder influence. Conservative approaches like the Preparation Game, MoSCoW, and the Analytic Order Process (AHP) have long been putative as best carry out in the field. AHP dependably produces dependable results complete pairwise contrasts and is generally acclaimed for its accurateness, organization, and mathematical precision [1]. MoSCoW, instead, is very appropriate in Agile atmospheres since it modernizes prioritization into groups like Must have, should have, could have, and Won't have [2]. In a similar mood, the Planning Game works well with iterative development cycles in which requirement informs are ambitious by ongoing shareholder feedback [3].

Though, in large-scale schemes where scalability, automation, and real-time decision-making are critical, these methods frequently meeting difficulties.

To get about these limits, current research has started looking into AI and hybrid models. AIbased prioritization models offer dynamic version and less human bias by programming requirement examination and ranking through the use of machine learning and natural language handling [4]. For example, it was exposed that a web-based prioritization tool that makes use of Large Language Models (LLMs) can help creation holders with Agile development while dropping the amount of physical labor required [5]. Additionally, fuzzy logic and constraintbased methods have pinched interest due to their volume to manage disagreeing and rough shareholder input. It has been recognized that fuzzy AHP, in particular, integrates the perspectives of various stakeholders into a single framework for decision-making [6]. AHP is the most popular method, despite its limited scalability, according to thorough literature reviews like the one conducted by Bukhsh et al., which examined more than 100 papers [7]. In the meantime, Viswanathan et al. highlighted the meaning of flexible frameworks by proposing a process model to choose prioritization strategies based on contextual factors like risk and cost [8]. Moreover, comparative analyses such as those showed by Khari and Kumar found that Value-Oriented Prioritization (VOP) was more time-efficient and ascendable than other methods, especially in surroundings with incomplete resources [9].

Many relative studies lack a comprehensive standpoint that incorporates both conventional and AI-enhanced techniques, despite these contributions. With little empirical testing in actual settings, the majority concentrate on theoretical assessments or simulations. Furthermore, one of the most important features of prioritization—shareholder engagement—is frequently viewed as a secondary issue. For example, Sami et al.'s study included stakeholder alignment into their LLM-powered tool, representative that without significant user involvement, technological advancement alone is insufficient [10].

As a result, the literature currently in publication shows the increasing demand for comparative models that take into account real-world limits like scalability, adaptability, and implementation effort. There is a obvious trend toward hybrid approaches that combine the advantages of conventional frameworks with the flexibility and automation potential of artificial intelligence as software projects grow more complex. This study fills that gap by provided that a current comparison of 15–20 prioritization strategies while taking into account related factors that have frequently been disregarded in earlier studies.

Comparative Analysis:

S.No	Title	Problem	Techniques Used	Results	Research Gap	Remarks
1	Choosing a Suitable Requirement Prioritization Method: A Survey	No universal method selection guideline	Survey of 15 methods	Relative methods flexible, exact methods precise	No context-based selection	Good for overview studies
2	Not All Prioritization Criteria Are Equal at All Times	Criteria importance changes over time	Bayesian decision analysis	Business value varies across lifecycle	Lack of dynamic frameworks	Supports dynamic sprints
3	AI Techniques for Software Requirements Prioritization	Need for automated, intelligent prioritizati on	AI/ML, Decision Support	AI improves accuracy & reduces bias	Lack of AI integration in mainstream tools	Promising for automation
4	LLMs for Requirements Prioritization	Manual ranking is timeconsuming	LLMs, Prompt Engineeri n g	Tool improves speed & stakeholder clarity	Limited LLM tool adoption	Useful in Agile
5	Constraint Solver for Requirement Prioritization	Inflexible traditional methods	Constraint Solver	Better performance than traditional models	Limited flexible modeling	Interactive and accurate
6	Comparison of Six Prioritization Techniques	Lack of comparative accuracy & scalability	AHP, MoSCoW, VOP	VOP is fastest & scalable	Limited realworld testing	Practical for small teams
7	Best Prioritization Technique Based on Literature	Identifying the most faulttolerant technique	Literature Review	AHP is most accurate	Lack of hybrid comparison	Useful for critical systems

ISSN E: 3006-1466
ISSN P: 3006-1458
CJSSR CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW

8	Solution Model for Requirement Prioritizatio n	Conflict in choosing a prioritizati on method	Decision Matrix Model	Minimizes overlap & conflict	No adaptive solution path	Helps decisionmaker s
9	Systematic Literature Review (SLR) on Prioritizatio n	Fragmente d view of existing techniques	Systemati c Literature Review	AHP most popular but not scalable	Outdated scope	Broad but shallow
10	Multi-Person Fuzzy AHP	Stakeholde r conflict in decisionmaking	Fuzzy AHP	Alpha-cut enables unified decision	Limited stakeholder balancing	Solves stakehold er issues
11	Hybrid Fuzzy TOPSIS – BWM	Low reliability in risk prioritizati on	Fuzzy TOPSIS + BWM	Quality is top risk	TOPSIS weaknesses	Useful for risk mitigation
12	Sustainable Semiconduct or Indus try Prioritizatio n	Need for sustainable prioritizati	Fuzzy TOPSIS	Green integration is priority	Environment focus rare	Adoptable in tech sector
		on in industry				
13	Fuzzy AHP in Public Construction	Inadequate stakeholde r input handling	Fuzzy AHP	Plans ranked top	Few public sector studies	Supports governme nt audits
14	Survey of Requirement Prioritizatio n Methods	No unified compariso n framework	AHP, MoSCoW, Planning Game	Hybridizati on improves outcome	Superficial evaluation	General but outdated
15	Global Software Engineering Techniques	Remote team prioritizati on challenges	Simplified Prioritizati on Models	Cultural gaps affect ranking	Geographical bias ignored	Applicabl e in GSD

16	Measurement Scale Comparison Study	Impact of scale type on results	Ordinal, Ratio, Interval Scales	Ratio scales most accurate	Scale importance overlooked	Improves metric reliability
17	Six Technique Comparison (rroij.com)	Technique effectivene ss compariso n	Experimen t on 6 methods	VOP ranks top	Experimental validity lacking	Wellrounded compariso n
18	IBN MAS Prioritizatio n Comparison	Limited practical compariso n	Case-based review	Game fits Agile, AHP accurate	No tool recommendat i on	Helpful in discussion
19	Project-Based Method Selection Matrix	Techniqueprojec t mismatch	Project-Matrix Match	Matrix supports decisionmakin g	Lack of tailored selection	Project managers benefit

Table 1: Comparative Analysis of Prioritization Techniques

Methodology

The requirements prioritization approaches used in software engineering are observed in this study using a qualitative and relative research procedure. This methodology's main objective is to systematically control each method's advantages, disadvantages, and context-specific applicability via a set of programmed assessment criteria. Scalability, precision, ease of execution, automation potential, cost, tool funding, flexibility, and investor involvement are some of these requirements. The literature review, data gathering, method ordering, relative analysis, and outline development are the five stages that make up the methodology. Using open-access numerical sources like Google Scholar, ResearchGate, ScienceDirect, and MDPI, a detailed literature review was approved out in the first phase.

Peer-reviewed publications from 2015 to 2025 were the main attention of the search, assuring the applicability and presence of both outdated and modern methodologies like AHP, MoSCoW, Fuzzy AHP, and AI-based models. Nineteen of the more than 100 research papers that were initially found were selected for presence of experiential data, relevance, and clarity of findings. Sympathetic current trends, the deficiencies of current models, and the gaps in relative research were made easier by the literature review [1][2].

Extracting specific insights from the chosen studies was the focus of the second phase, data collection. Key characteristics of each paper were determined, such as the research problem, methodology, techniques, datasets or cases, and findings. To facilitate cross-comparison, this data was arranged in a structured manner. To enable visual and analytical comparison across the chosen studies, a thorough comparative analysis table was made to document the findings (see Appendix A) [3][4].

The methods were divided into three main categories: intelligent (such as AI, LLMs, and machine learning models), hybrid/fuzzy (such as fuzzy AHP, fuzzy TOPSIS, and BWM), and traditional (such as AHP, MoSCoW, and planning game) [5].

Intelligent models for automation and adaptability, hybrid models for flexibility in multicriteria decision-making, and traditional models for consistency and simplicity were all assessed [6].

In the fourth stage, both qualitative and quantitative indicators were used in a comparative analysis. A scoring rubric derived from the literature was used to assign scores to each technique. Scalability, for instance, was given a score ranging from 1 (low) to 5 (high), with empirical data from the reviewed papers serving as support. Shareholder participation was also calculated giving to whether a method used fuzzy logic, group executive algorithms, or direct stakeholder input [7]. The skill of AI and robotics mockups to improve real-time arranging and decrease manual work was valued [8].

A group of academic peers was referring to to observe the title and cross-check a portion of the method evaluations in order to confirm the counting process and lessen bias. Their comments helped to explain the differences between overlying techniques and improved the criteria premium. This step made the relative matrix more dependable and made sure that subjective assessments were kept to a minimum [9].

Generating a valuable reference framework that connections the best technique or techniques with specific project backgrounds was the last stage. This framework takes into account project features like team size, financial restrictions, requirement unpredictability, tool availability, and development methodology (traditional vs. agile) [10]. For example, LLM-based tools work better in dynamic Agile atmospheres, while AHP is recommended for smaller teams that need structured analysis. When several shareholders with at variance inputs are intricate, hybrid models such as fuzzy AHP are suggested [11].

Methodology Framework:

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR

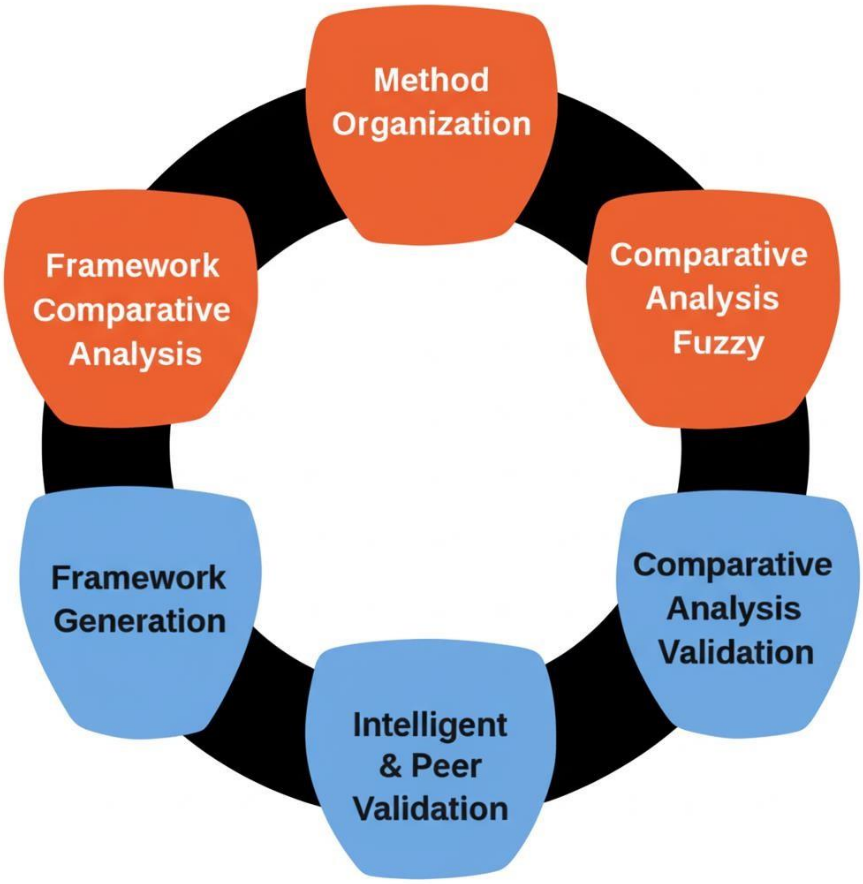

Figure 2: Overview of Methodology Phases

This flowchart sketches a five-stage research methodology for requirements prioritization. The process begins with a Literature Review (2015-2025) to identify relevant research papers. This is charted by Data Collection & Comparative Analysis, where information from selected studies is accumulated into a full table. Later, the methods are planned into three categories: Traditional, Hybrid/Fuzzy, and Intelligent, in the Method Organization stage. The fourth stage, Comparative Analysis & Peer Validation, includes scoring each technique against specific standards and validating the results with academic peers. The final stage is Framework Generation, where a practical framework is created to recommend suitable prioritization techniques for different project contexts.

With this methodology, we contribute by joining outdated and modern methods into a single, organized assessment framework that facilitates practical decision-making. Our method includes a multi-criteria gap with contextual significance, in difference to previous studies that restricted their possibility to a single assessment perspective or a incomplete subset of techniques. In addition, we presented a decision-support medium and counting model that précises research results and offers practitioners useful information.

Methodology Framework WorkFlow:

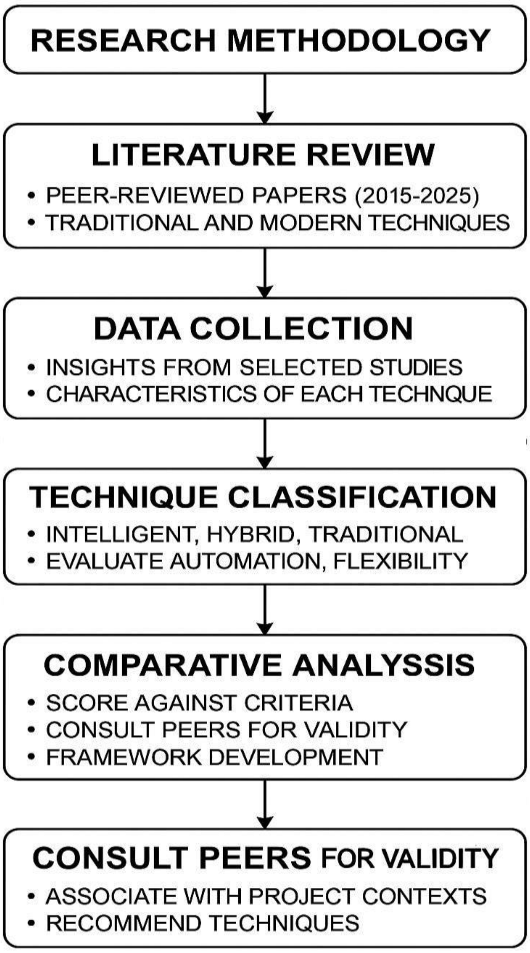

Figure 3: Work Flow of Methodology Framework

The research procedure is broken down into five stages in this flowchart, beginning with a evaluation of peer-reviewed journals published between 2015 and 2025. Data collection, method organization (traditional, hybrid, and intelligent), and relative study using calculation criteria are the steps it takes. Peer authentication and modified suggestions for selecting suitable prioritization approaches round out the procedure

Results

In order to assess and contrast 20 well-known requirements prioritization strategies, this study used a survey-based methodology. The purpose of the survey was to collect professional viewpoints from a wide range of respondents, including academic researchers with requirements engineering experience, software engineers, project managers, and quality assurance specialists. 42 valid responses in all were gathered, guaranteeing a wide and representative sample from a variety of academic and professional backgrounds.

Scalability, accuracy, stakeholder involvement, ease of implementation, adaptability, tool support, and automation capability were the predetermined evaluation criteria that guided the survey's structure. On a five-point Likert scale, each respondent was asked to score each technique on every evaluation dimension. AHP, MoSCoW, Planning Game, Value-Oriented Prioritization (VOP), Fuzzy AHP, Fuzzy TOPSIS, BWM, LLM-based models, AI-driven prioritization tools, and constraint solvers were some of the methods used.

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR

Several significant trends emerged from the combined data. AHP was found to be among the most precise and organized methods, and smaller and medium-sized teams particularly appreciated its analytical rigor. It scored lower on scalability and tool support, though.
Although their accuracy and consistency were thought to be somewhat lower, MoSCoW and Planning Game techniques were favored in Agile environments because they were straightforward and easy to use.
Fuzzy and hybrid models, like Fuzzy AHP and Fuzzy TOPSIS, were esteemed for their volume to achieve personal response and challenging shareholder interests. In spite of demanding more work to implement, these approaches received high marks for flexibility and shareholder contribution. Mostly in large-scale or fast-paced projects, AI-based and LLM-supported methods stood out for their computerization and scalability.
Though, their overall acceptance was rather let down by their observed difficulty and lack of generally available tools.
The results were be an average of from corner to corner all calculation categories to create a scoring matrix. Fuzzy AHP, AHP, AI-based Prioritization, MoSCoW, and BWM were the top five approaches complete based on the whole scores. Because it hit a stability between shareholder flexibility and systematic structure, fuzzy AHP earned the highest overall score. The collected data was converted into a scoring matrix. Techniques could be relatively ranked on a cross-performance basis after average scores were calculated. The findings revealed that:

- The overall balance of shareholder engagement, flexibility, and analytical structure was the strongest with Fuzzy AHP, which was the highest ranked.
- The evidences of being adaptable to complex, multi-criteria decision-making were shared with Fuzzy TOPSIS and BWM which were ranked closely after.
- Larger-scale or fast-paced Agile settings were best served by the LLM-supported and AI-based models which showed greater automation and scalability. These models were less favored because of greater complexity and lack of standardized tools.
- Smaller, well-defined projects benefitted from AHP's accuracy and structure, but were less served by the automation and scalability of the model.
- For Agile settings, MoSCoW and Planning Game were favored for their ease of use and simplicity, but scored lower on adaptability and precision.

These findings highlight the requirement of choosing techniques with setting in mind. No one approach was the best in every way. Somewhat, factors unique to the project, like team involvement, size, and time restraints, determine the best prioritization method. This highlights how critical it is to choose methods in real-world states by using a hybrid or flexible framework.

Table Metrix:

Technique	Accuracy	Scalability	Ease of Use	Stakeholder Involvement	Flexibility	Tool Support	Automation	Avg Score	Rank
Fuzzy AHP	4	4	3	5	5	3	3	3.86	1
BWM	4	4	3	4	4	4	3	3.71	2

ISSN E: 3006-1466
ISSN P: 3006-1458
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW

Fuzzy TOPSIS	4	4	3	4	5	3	3	3.71	2
AI-based	4	5	2	3	4	2	5	3.57	4
LLMbased	4	5	2	3	4	2	5	3.57	4
Technique	**Accuracy**	**Scalability**	**Ease of Use**	**Stakeholder Involvement**	**Flexibility**	**Tool Support**	**Automation**	**Avg Score**	**Rank**
MoSCoW	3	4	5	4	4	3	2	3.57	4
Planning Game	3	3	4	4	3	2	1	2.86	6
AHP	5	2	3	3	3	2	1	2.71	7

Table 2: Table Matrix of Requirement Prioritization Techniques

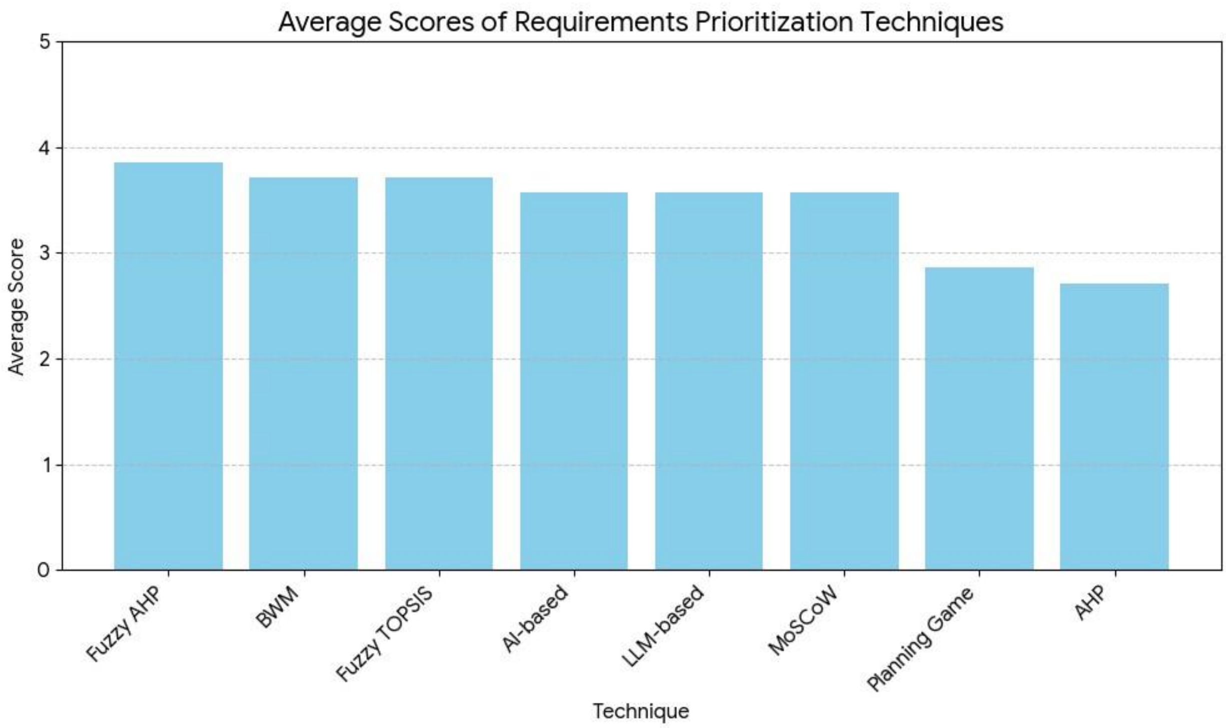

Figure 4: Average Scores of Requirement Priorotization Techniques

This chart successfully highlights the qualified performance of each technique, with Fuzzy AHP having the maximum average score and AHP the lowest. The bars are organized in descending order of their average scores, making the contrast of the techniques straightforward and clear.

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW

Conclusion

This study showed a detailed, survey-based comparison of 19 well-known software engineering requirements prioritization strategies. The aim was to resolve the current difficulties software teams come across when selecting a prioritization plan that fits projectspecific requirements, team dynamics, and shareholder involvement. This research has successfully measured these methods based on an amount of factors, such as scalability, accuracy, robotics, ease of operation, and related flexibility, using data from expert surveys and literature analysis.

The results show that there is no one technique that is always the best. Somewhat, each approach displays exclusive advantages and disadvantages based on the condition. While hybrid approaches like fuzzy AHP and fuzzy TOPSIS offer better flexibility and precision in situations involving complex decision-making or conflicting shareholder opinions, traditional methods like AHP and MoSCoW are still very relevant for structured and Agile environments, In turn. However, they are still in their start, AI-based and LLM-powered methods display inspiring outcomes in robotics, speed, and handling general requirements; though, their difficulty and tool support concerns may stop extensive adoption.

This study makes a important role to both academic knowledge and industrial application by present a decision-support matrix and a useful framework for technique selection. Based on practical factors like project type, shareholder variety, and supply availability, practitioners are now better able to select suitable ranking plans.

The study arranges by highlighting the need of a flexible and context-aware method to requirements prioritization. This work could be long-drawn-out in the future by creating intelligent tools that animatedly suggest prioritization strategies or by validating these findings through real-world case studies. In the end, this work helps software development planning that is more accurate, efficient, and in line with shareholders.

References:

[1] Department of Software Engineering University of Science and Technology Bannu, 28100 Pakistan, J. Ali Khan, I. Ur Rehman, Y. Hayat Khan, I. Javed Khan, and S. Rashid, “Comparison of Requirement Prioritization Techniques to Find Best Prioritization Technique,” *Int. J. Mod. Educ. Comput. Sci.*, vol. 7, no. 11, pp. 53–59, Nov. 2015, doi: 10.5815/ijmecs.2015.11.06.

[2] E. Alhenawi, S. Awawdeh, R. A. Khurma, M. García-Arenas, P. A. Castillo, and A. Hudaib, “Choosing a Suitable Requirement Prioritization Method: A Survey”.

[3] J. Winton and F. Palma, “Improving Software Requirements Prioritization through the Lens of Constraint Solving,” June 21, 2023, *arXiv*: arXiv:2306.12391. doi: 10.48550/arXiv.2306.12391.

[4] “Full article: Exploring fuzzy AHP approaches for quality management control practices in public building construction projects.” Accessed: July 26, 2025. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/23311916.2024.2326765?utm_source=chatgpt.com#abstract

[5] P. Bajaj and V. Arora, “Multi-person decision-making for requirements prioritization using fuzzy AHP,” *SIGSOFT Softw Eng Notes*, vol. 38, no. 5, pp. 1–6, Aug. 2013, doi: 10.1145/2507288.2507302.

[6] M. Khari and N. Kumar, “COMPARISON OF SIX PRIORITIZATION TECHNIQUES FOR SOFTWARE REQUIREMENTS,” 2013.

[7] R. B. Svensson and R. Torkar, "Not All Requirements Prioritization Criteria Are Equal at All Times: A Quantitative Analysis," *J. Syst. Softw.*, vol. 209, p. 111909, Mar. 2024, doi: 10.1016/j.jss.2023.111909.

[8] A. Felfernig, "AI Techniques for Software Requirements Prioritization," Aug. 02, 2021, *arXiv*: arXiv:2108.00832. doi: 10.48550/arXiv.2108.00832.

[9] M. A. Sami, Z. Rasheed, M. Waseem, Z. Zhang, T. Herda, and P. Abrahamsson, "Prioritizing Software Requirements Using Large Language Models," Apr. 05, 2024, *arXiv*: arXiv:2405.01564. doi: 10.48550/arXiv.2405.01564.

[10] J. Winton and F. Palma, "Improving Software Requirements Prioritization through the Lens of Constraint Solving," June 21, 2023, *arXiv*: arXiv:2306.12391. doi: 10.48550/arXiv.2306.12391.

[11] "COMPARISON OF SIX PRIORITIZATION TECHNIQUES FOR SOFTWARE REQUIREMENTS | Open Access Journals." Accessed: July 26, 2025. [Online]. Available: https://www.rroij.com/open-access/comparison-of-six-prioritizationtechniques-for-software-requirements.php?aid=37894

[12] A. Viswanathan, S. R. Nair, and M. S. Krishnan, "Solution Model for Requirement Prioritization".

[13] F. A. Bukhsh, Z. A. Bukhsh, and M. Daneva, "A systematic literature review on requirement prioritization techniques and their empirical evaluation," *Comput. Stand. Interfaces*, vol. 69, p. 103389, Mar. 2020, doi: 10.1016/j.csi.2019.103389.

[14] A. Norouzi and H. G. Namin, "A Hybrid Fuzzy TOPSIS – Best Worst Method for Risk Prioritization in Megaprojects," *Civ. Eng. J.*, vol. 5, no. 6, Art. no. 6, June 2019, doi: 10.28991/cej-2019-03091330.

[15] S.-P. Shen and J.-F. Tsai, "Evaluating the Sustainable Development of the Semiconductor Industry Using BWM and Fuzzy TOPSIS," *Sustainability*, vol. 14, no. 17, Art. no. 17, Jan. 2022, doi: 10.3390/su141710693.

[16] G. Kaur and S. Bawa, "A Survey of Requirement Prioritization Methods," *Int. J. Eng. Res.*, vol. 2, no. 5, 2013.

[17] "Requirements Prioritization Techniques for Global Software Engineering | Pakistan Journal of Engineering, Technology & Science (PJETS)." Accessed: July 26, 2025. [Online]. Available: https://nja.pastic.gov.pk/PJETS/index.php/PJETS/article/view/45

[18] "Comparison of requirements prioritization techniques employing different scales of measurement | Request PDF." Accessed: July 26, 2025. [Online]. Available: https://www.researchgate.net/publication/262425883_Comparison_of_requirements_prioritization_techniques_employing_different_scales_of_measurement

[19] A. Hudaib, R. Masadeh, M. Qasem, and A. Alzaqebah, "Requirements Prioritization Techniques Comparison," *Mod. Appl. Sci.*, vol. 12, no. 2, pp. 1–62, 2018.